I claim:

Sub a7

1. A computerized software installation and installation validation system comprising:

a database engine module for maintaining a first database, said first database having a first custom action table, said first custom action table having a plurality of action columns and an action row; and

an installation engine module operative to read an action value from the action column of the action row and causing an action specified by the action value to be performed by a computer.

2. The system of claim 1, wherein the action value specifies an executable program.

3. The system of claim 1, wherein the action value specifies a dynamically loadable module.

4. The system of claim 1, wherein the action value specifies a script.

5. The system of claim 4, wherein the script is in a JavaScript syntax or a Visual Basic syntax.

6. The system of claim 1, wherein the first database further includes a first sequence table specifying an order in which to process the action.

7. A computerized software installation and installation validation system comprising:

a database engine module for maintaining a product database and a validation database, said validation database having a custom action table, said custom action table having a plurality of action columns and an action row; and

an installation engine module operative to merge the product database and the validation database into a merged table set and to read an action value from the action column of the action row within the custom action table of the merged table set and causing an action specified by the action value to be performed by a computer.

8. The system of claim 7 wherein the table set is written to a persistent database.

Sub a3

9. A method for an installation program to install software comprising:
identifying an action stored within a database;
reading the action from the database;
identifying a type for the action;
executing the action; and
communicating the results of the action to the installation program.

10. The method of claim 9, wherein the type of the action specifies a dynamically loadable library and executing the action loads a module from the dynamically loadable library image.

11. The method of claim 9, wherein the type of the action specifies an executable program and executing the action loads the program and causes it to execute.

12. The method of claim 9, wherein the type of the action specifies a script and executing the action submits the script to an interpreter.

13. The method of claim 12, wherein the script comprises instructions in the JavaScript language or the Visual Basic language.

14. The method of claim 9, wherein executing the action causes the action to be stored in a table of actions to be deferred until a commit action is executed.

Sub a4

15. A computer comprising:

a processor;

a computer-readable medium;

a database engine module for maintaining a first database, said first database having a first custom action table, said first custom action table having a plurality of action columns and an action row; and

an installation engine module operative to read an action value from the action column of the action row and causing an action specified by the action value to be performed by a computer.

16. The computer of claim 15, wherein the action value specifies an executable program.

17. The computer of claim 15, wherein the action value specifies a dynamically loadable module.

18. The computer of claim 15, wherein the action value specifies a script.

19. The computer of claim 18, wherein the script is in a JavaScript syntax or a Visual Basic syntax.

Sub a5

20. A computer readable medium having computer-executable instructions for performing the steps comprising:

identifying an action stored within a database;

reading the action from the database;

identifying a type for the action;

executing the action; and

communicating the results of the action to the installation program.

21. The medium of claim 20, wherein the type of the action specifies a dynamically loadable library and executing the action loads a module from a dynamically loadable

library image.

22. The medium of claim 20, wherein the type of the action specifies an executable program and executing the action loads the program and causes it to execute.

23. The medium of claim 20, wherein the type of the action specifies a script and executing the action submits the script to an interpreter.

24. The medium of claim 23, wherein the script comprises instructions in the JavaScript language or the Visual Basic language.

25. The medium of claim 20, wherein executing the action causes the action to be stored in a table of actions to be deferred until a commit action is executed.

26. A computer-readable medium having stored thereon a data structure comprising:

a first data field containing data specifying a name of an action;

a second data field containing data specifying a source for the action;

a third data field containing data specifying a target for the action; and

a fourth data field containing data specifying a type of the action, wherein during a predetermined data processing operation the action is read from the source specified by the second data field and submitted to an operating system component determined by the fourth data field and wherein the action is started from a location determined by the third data field.

Add
a4